Towards a Provenance-Aware Internet of Things (IoT) System

Ebelechukwu Nwafor [*], Gedare Bloom[*], Andre Campbell[*] and David Hill[*]
[*]Department of Electrical Engineering and Computer Science
Howard University, Washington, DC 30332–0250
Email: see ebelechukwu.nwafor@bison.howard.edu

Abstract—**The Internet of Things (IoT) offers immense benefits by enabling devices to leverage networked resources theryby making intelligent decisions. The numerous heterogeneous connected devices that exist throughout the IoT system creates new security and privacy concerns. Some of these concerns can be overcome through trust, transparency, and integrity, which can be achieved with data provenance. Data provenance, also known as data lineage, provides a history of transformations that occurs on a data object from the time it was created to its current state. Data provenance has been explored in the areas of scientific computing, business, forensic analysis, and intrusion detection. Data provenance can help in detecting and mitigating malicious cyber attacks. In this paper, we explore the integration of provenance within the IoT. We propose a provenance collection framework for IoT applications. We evaluate the effectiveness of our framework by looking at an application of provenance data by developing a prototype system for proof of concept.**

I. Introduction

The Internet of Things (IoT) has generated a lot of buzz among commercial and industrial information technology experts all over the world. Heterogeneous devices like we have never seen before are communicating with each other over a shared network (e.g internet). For example, It is possible to automatically control the temperature of a house remotely though a cell phone. With this unprecedented communication, here has been an exponential increase in the number of devices connected to the internet. It is estimated that over 50 million devices will be connected to the internet by the year 2020. With the vast amounts of connected heterogeneous devices, security and privacy risks is increased. IoT devices are not strongly incorporated with security in mind. This raises the complexity of including security after the device has been deployed. For instance some devices come with default passwords which might never be changed during its lifecycle. There is a strong need to provide integrity to data produced IoT devices. One of the ways of achieving this is through which can be addressed through data provenance. Data provenance is a comprehensive history of activities that occurred on an entity from its origin to its current state. Provenance ensures integrity of data . Provenance has been applied in various area such as scientific workflow for experiment reproducibility, and information security as a form of access control and also for intrusion detection in mitigating malicious adversaries. Provenance ensure trust and integrity of data. IoT devices (things) produces sensor-actuator data. A workflow representation of of how sensor data is generated can be generated to depict dependency between sensor-actuator readings and devices/sensor information contained in the device.

This information generated can be prove to be beneficial as a means for mitigating malicious intrusion or for scientific reproducability as provenance. In this paper, we propose a provenance aware framework for IoT devices, in which provenance data is collected and modeled to represent dependencies between sensor-actuator readings and the various entities contained in the IoT architecture. The remaining sections of is divided as follows: section 2 discusses background information on IoT definition, architecture ,application domain and data provenance. Section 3 discusses the need for a provenance framework using a use case scenario of an automated smart home. Section 3 talks about related work in provenance collection systems. Section 4 discusses implementation details for provenance collection framework. Section 5 talks about results and experiment analysis. Finally, in section 6 we conclude with future work.

II. Background

This section describes key concepts of data provenance, IoT characteristics, and provenance models. It also provides motivating example for the need for provenance collection via a use case.

A. Internet of Things

There is no standard definition for IoT, however, researchers have tried to define the concept of connected ̈things ̈. The concept of IoT was proposed by Mark Weiser in the early 1990s which represents a way in which the physical objects, ̈things ̈, can be connected to the digital world. Gubbi et al defines the IoT as an interconnection of sensing and actuating devices that allows data sharing across platforms through a centralized framework. We define (IoT) as follows: The Internet of Things (IoT) is a network of heterogeneous devices with sensing and actuating capabilities communicating over the internet.

The notion of IoT has been attributed to smart devices. The interconnectivity between various heterogeneous devices allows for devices to share information in a unique manner. Analytics is a driving force for IoT. With analytics, devices can learn from user data to make smarter decisions. This notion of smart devices is seen in various commercial applications such as smartwatches, thermostats that automatically learns a user patterns. The ubiquitous nature of these devices make them ideal choices to be included in consumer products. IoT architecture represents a functional hierarchy of how information is disseminated across multiple hierarchies contained in an IoT framework; from devices which contain sensing and actuating capabilities to massive data centers (cloud storage). Knowing how information is transmitted across layers allows a better understanding on how to model the flow of information across actors contained in an IoT hierarchy. Figure 1 displays the IoT architecture and the interactions between the respective layers. IoT architecture consists of four distinct layers: The sensor and actuator layer, device layer, gateway layer and the cloud layer. The base of the architectural stack consist of sensors and actuators which gathers provenance information and interacts with the device layer. The device layer consists of devices (e.g mobile phones, laptops, smart devices) which are responsible for aggregating data collected from sensors and actuators. These devices in turn forwards the aggregated data to the gateway layer. The gateway layer routes and forwards data collected from the device later. It could also serve as a medium of temporary storage and data processing. The cloud layer is involved with the storage and processing of data collected from the gateway layer. Note that the resource constraints decreases up the architectural stack with the cloud layer having the most resources (memory, power computation) and the sensor- actuator layer having the least.

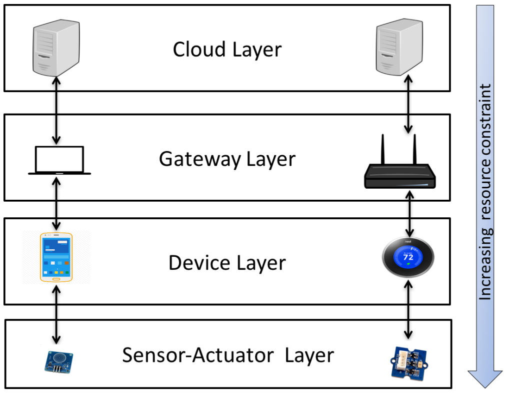

Fig. 1. IoT Architecture Diagram. The arrows illustrates the interaction between data at various layers on the architecture.

With the recent data explosion [22] due to the large influx in amounts of interconnected devices, information is disseminated at a fast rate and with this increase involves security and privacy concerns. Creating a provenance-aware system is beneficial to IoT because it ensures the trust and integrity of interconnected devices. Enabling provenance collection in IoT devices allows these devices to capture valuable information which enables backtracking in an event of a malicious attack.

B. Provenance-Aware IoT Device Use Case

Consider a smart home as illustrated in Figure 2 that contains interconnected devices such as a thermostat which automatically detects and regulates the temperature of a room based on prior information of a user's temperature preferences, a smart lock system that can be controlled remotely and informs a user via short messaging when the door has been opened or is closed, a home security camera monitoring system, a smart fridge which sends a reminder when food products are low. In an event that a malicious intruder attempts to gain access to the smart lock system and security camera remotely, provenance information can be used to track the series of events to determine where and how a malicious attack originated. Provenance can also be used as a safeguard to alert of a possible remote or insider compromise thereby protecting against future or ongoing malicious attacks.

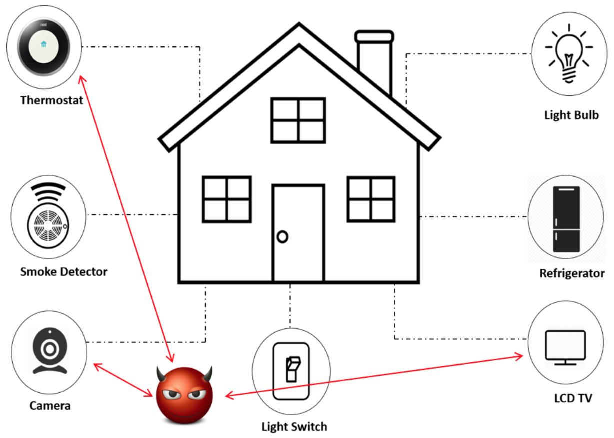

Fig. 2. Smart home use case Diagram

C. Data Provenance

The Oxford English dictionary defines provenance as the place of origin or earliest known history of something". An example of provenance can be seen with a college transcript. A transcript is the provenance of a college degree because it outlines all of the courses satisfied in order to attain the degree. In the field of computing, data provenance, also known as data lineage, can be defined as the history of all activities performed on entities from its creation to its current state. Cheney et al. decribes provenance as the origin and history of data from its lifecycle. Buneman et al describes provenance from a database perspective as the origin of data and the steps in which it is derived in the database system. We formally define provenance as follows: Data provenance of an entity is a comprehensive history of activities that occur on that entity from its creation to its present state.

Provenance ensures trust and integrity of data [**?**]. It outlines dependency between all objects involved in the system and allows for the verification of the source of data. dependency is used to determine the relationship between multiple objects. The relationship in which provenance denotes can in turn be used in digital forensics [**?**] to investigate the cause of a malicious attack and also in intrusion detection systems to further enhance the security of computing devices.

Provenance has been utilized in application domains such as computer security for access control and intrusion detection to detect malicious activities, in scientific experiments for reproducability and in version control systems to mention but a few.

D. Provenance Characteristics

Since provenance denotes the who, where and why of data transformation, it is imperative that data disseminated in an IoT architecture satisfies the required conditions. The characteristics of data provenance are outlined in detail below.

- Who: This characteristic provides information on activities made to an entity. Knowing the "who" characteristic is essential because it maps the identity of modification to a particular data object. An example of "who" in an IoT use case is a sensor device identifier.
- Where: This characteristic denotes location information in which data transformation was made. This provenance characteristic could be optional since not every data modification contains location details.
- When: This characteristic denotes the time information at which data transformation occurred. This is an essential provenance characteristic. Being able to tell the time of a data transformation allows for tracing data irregularities.
- What: This characteristic denotes the transformation is applied on a data object. A use case for IoT can be seen in the various operations (create, read, update, and delete) that could be performed on a file object.

E. Model for representing provenance for IoT

In order to represent the right kind of provenance information, we need to satisfy the who, where, how, and what of data transformations. Provenance data can be represented using a provenance model in a modeling language such as, PROVDM which is represented in serialized form as a JSON object. This model displays the causal relationship of data objects. We propose a model that contains information such as sensor readings, device name, and device information. There are two widely accepted modeling languages for representing provenance, PROV-DM [?] and Open Provenance Model [?] that have been applied in various literature and are considered standard for representing provenance. Details on the provenance models are outlined below.

F. Provenance Data Model (Prov-DM)

PROV-DM is a W3C standardized extension of OPM. Prov-DM is a model that is used to depict causal relationships between entities, activities and agents (digital or physical). It creates a common model that allows for interchange of provenance information between heterogeneous devices. It contains two major components: types and relations.

- Entity: An entity is a physical or digital object. An example of an entity is a file system, a process, or an motor vehicle. An entity may be physical or abstract.
- Activity: An activity represents some form of action that occurs over a time frame. Actions are acted upon by an agent. An example of an activity is a process opening a file directory, Accessing a remote server.
- Agent: An agent is a thing that takes ownership of an entity, or performs an activity. An example of an agent is a person, a software product, or a process.

The figure below illustrates the various types contained in PROV-DM and their representation. Entities, activities and agents are represented by oval, rectangle and hexagonal shapes respectively.

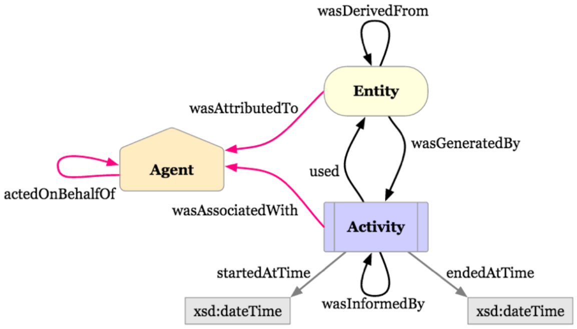

Fig. 3. Prov-DM respresentation showing various types contained in the model (Entity, Activity, and Agent)

Similar to the OPM, PROV-DM does not keep track of future events. PROV-DM relations are outlined below:

- wasGeneratedBy: This relation signifies the creation of an entity by an activity.
- used: This relation denotes that the functionality of an entity has been adopted by an activity.
- wasInformedBy: This relation denotes a causality that follows the exchange of two activities.
- wasDerivedFrom: This relation represents a copy of information from an entity.
- wasAttributedTo: This denotes relational dependency to an agent. It is used to denote relationship between entity and agent when the activity that created the agent is unknown.
- wasAssociatedWith: This relation denotes a direct association to an agent for an activity that occurs. This indicates that an agent plays a role in the creation or modification of the activity.
- actedOnBehalfOf: This relation denotes assigning authority to perform a particular responsibility to an agent. This could be by itself or to another agent.

Some of the difference between OPM and PROV-DM are described below:

- The main components Artifact, Process and Agent in the OPM model are modified to Entity, Action, and Agent.
- Additional causal dependencies such as wasAttributedTo and actedOnBehalfOf were included to represent direct and indirect causal dependencies respectively between agents and entities.

III. Conclusion

The conclusion goes here.

Acknowledgment

The authors would like to thank...

References

[1] H. Kopka and P. W. Daly, *A Guide to LaTeX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.